



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/660,034	09/11/2003	Cary Lee Bates	ROC920030168US1	5835

7590 12/01/2006  
Grant A. Johnson  
IBM Corporation - Dept. 917  
3605 Highway 52 North  
Rochester, MN 55901

EXAMINER

INGBERG, TODD D

ART UNIT PAPER NUMBER

2193

DATE MAILED: 12/01/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

# Office Action Summary

Application No.

10/660,034

Applicant(s)

BATES ET AL.

Examiner

Todd Ingberg

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 06 September 2006.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-21 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-21 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 9/6/06 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

Art Unit: 2193

### DETAILED ACTION

Claims 1 – 21 have been examined.

Claims 1, 3, 8, 9, 14, 17 and 18 have been amended.

#### *Drawings*

1. Drawing changes have been accepted..

#### *Specification*

2. The new Title and new Abstract have been accepted.

#### *Claim Rejections - 35 USC § 101*

3. Rejected under 35 U.S.C. 101 has been overcome by amendment.

#### *Claim Rejections - 35 USC § 112*

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claim 1 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Please, review the following sentence in claim 1.

an apparatus for implementing enhanced graphical user interface functions in a graphical debugger in a computer system, said graphical debugger for identifying errors in a program under debug , said graphical debugger including instructions stored on a computer readable storage medium, said instructions when executed by the computer system to cause the computer system implement functions . It seems awkward.

***Claim Rejections - 35 USC § 102***

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1- 21 are rejected under 35 U.S.C. 102(b) as being anticipated by “How Debuggers Work”, by Jonathan B. Rosenberg published 1996 (**Dbug**).

**Claim 1**

**Dbug** anticipates an apparatus for implementing enhanced graphical user interface functions in a graphical debugger in a computer system, said graphical debugger for identifying errors in a program under debug (debugger by definition), said graphical debugger including instructions stored on a computer readable storage medium, said instructions when executed by the computer system to cause the computer system implement functions (Dbug, page 22 – 32, Figures show enhanced GUI) comprising:

a user interface for operatively controlling a graphical user interface (Dbug, pages 22-32, the ability to perform operations such as disassembly view on page 29) ;

a loadmap display manager coupled to said user interface for implementing a loadmap function; said loadmap display manager generating a list of program objects being bound to the program under debug at runtime; (Dbug, page 236, anticipates debugging object oriented code – Page 2 of Applicant’s Specification loadmap is Admitted prior art as part of object oriented debugging and Debug, page 81);

said user interface responsive to said loadmap display manager for displaying a program loadmap (Dbug, page 94, JAVA debugging) including said generated list of program objects being bound to the program under debug at runtime.

a custom record display manager coupled to said user interface receiving user inputs and implementing a custom record display function (Dbug, page 22 – 32, Figures show enhanced GUI);

said user interface responsive to said custom record display manager for displaying user selected customized records (Dbug, pages 22-32, the ability to perform operations such as Inspector on page 32).

**NOTE**

The Examiner is interested in hearing how the added limitations are not inherent.

1. Debuggers are tools to identify and correct errors.
2. The programs loaded are the ones that are specified to be run with the debugger at runtime.

Art Unit: 2193

**Claim 2**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 1 further includes a debugger server and wherein said loadmap display manager operatively controls said debugger server for implementing said loadmap function. (Dbug, pages 195 – top of 198, remote debugging - page 197 for server specifically).

**Claim 3**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 2 wherein said load map function includes debugger means reading debug data for the program under debug (As per claim 1); examining the program debug data, for generating said list of each source and disassembly file contained in the program under debug (as per claim 1 – disassembly example), **and for generating said list of** ( As per claim 1 – JAVA example) program objects bound to the program under debug at run time to generate said program loadmap for display. As per claim 1.

**Claim 4**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 3 wherein said load map function further includes debugger means for identifying program load and unload events, and for dynamically updating said program loadmap for display as loadmap information changes responsive to program load and unload events (As per claim 1) ; whereby said program loadmap enables setting a breakpoint within a user selected address range for one instance of a source file, without setting the breakpoint in other instances of the source file (Dbug, Multithreaded Debugging in Chapter 9. And isolating specific threads on page 182 and Breakpointing on page 184)

**Claim 5**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 2 wherein said custom record display manager operatively controls said debugger server for implementing said custom record display function (as per claims 1 and 2); and said custom record display function includes debugger means for identifying a user selected variable (Dbug, page 32, Inspector – single variable), and user selected fields of the variable to be displayed (Dbug, page 32, Inspector – multiple as in display)

**Claim 6**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 5 wherein said custom record display function further includes debugger means for identifying a user selected all variables of this type to be customized (Dbug, page 130, watchpoints), for creating a custom record with the user selected fields and for adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields (Dbug, Chapter 6, page 108 – 110).

**Claim 7**

Apparatus for implementing enhanced graphical user interface functions as recited in claim 5 wherein said custom record display function further includes debugger means for creating a

Art Unit: 2193

custom record with the user selected fields and for adding said created custom record to a variables list for displaying the variable only with the user selected fields. As per claim 5.

**Claim 8**

A debugger computer program product for implementing enhanced graphical user interface functions in a computer system, said debugger computer program product for identifying errors in a computer program under debug and including instructions stored on a computer readable medium, said instructions when executed by the computer system to cause the computer system to perform the steps of:

- reading debug data for a program under debug;
- examining said program debug data, generating a list of each source (Dbug, pages 12 – 14) and disassembly file contained in the program under debug (Dbug, pages 29 and 146), and generating a list of each program object being (As per claim 3) bound to the program under debug at run time to generate said program loadmap for display; and
- displaying said generated program loadmap on a graphical user interface ; and dynamically updating said program loadmap for display as loadmap information changes during run time of the program under debug. ( as Per claim 1).

**Claim 9**

A debugger computer program product-for implementing enhanced graphical user interface functions as recited in claim 8 includes the step of identifying program load and unload events (Dbug, page 121, Step-Into), and dynamically updating said program loadmap for display responsive to program load and unload events (Dbug, update of display with Step operations is inherent and required).

**Claim 10**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 9 wherein the step of identifying program load and unload events includes the steps of setting breakpoints at program load and unload entry points in the program under debug (Dbug, page 108, bullet 7); and when debugging said program, identifying an unload breakpoint; and removing all information for an archive program being unloaded from said program loadmap to provide a current program loadmap, and displaying said current program loadmap. (Dbug, page 97, module unloaded).

**Claim 11**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 8 includes the step of identifying a user selected variable and identifying user selected fields of the variable to be displayed, and creating a custom record with the user selected fields. As per claim 5.

**Claim 12**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 11 includes the step responsive to a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying

Art Unit: 2193

user selected customized records for all variables of this type only with the user selected fields. As per claim 6.

**Claim 13**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 11 includes the step adding said created custom record to a variables list for displaying the variable only with the user selected fields. As per claim 7.

**Claim 14**

A debugger computer program product for implementing enhanced graphical user interface functions in a computer system, said debugger computer program product

**for identifying errors in a program under debug and**

including instructions executed by the computer system to cause the computer system to perform the steps of:

**generating a list of program objects being bound to the program under debug at runtime**  
**displaying a program loadmap including said generated list of program objects;**

identifying a user selected variable and identifying user selected fields  
of the variable to be displayed,  
creating a custom record with the user selected fields; and  
displaying the variable only with the user selected fields. As per claims 1 and 6.

**Claim 15**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 14 wherein the step of displaying the variable only with the user selected fields includes the step adding said created custom record to a variables list for displaying the variable only with the user selected fields. As per claim 3.

**Claim 16**

A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 14 includes the step of identifying a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields. As per claim 6.

**Claim 17**

A method for implementing enhanced graphical user interface functions in a graphical debugger **in a computer system, said graphical debugger identifying errors in a computer program under debug and including instructions stored on a computer readable storage medium, said instructions when executed by the computer system to cause the computer system to perform the steps of:**

utilizing a loadmap display manager for implementing a loadmap

Art Unit: 2193

function and generating a **list of program objects being bound to the program under debug at runtime to generate program** loadmap for the program under debug;

displaying said generated program loadmap responsive to said loadmap function;

utilizing a custom record display manager for receiving user inputs and implementing a custom record display function;

displaying user selected customized records responsive to said custom record display function. As per claim 1.

#### **Claim 18**

A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 17 wherein the step of utilizing a loadmap display manager for implementing a loadmap function and generating a program loadmap for a program under debug includes the steps of reading debug data for the program under debug; examining said program debug data, generating a list of each source and disassembly file contained in the program under debug, and generating a list of each program object bound to the program under debug at run time to generate said program loadmap for display; and displaying said generated program loadmap on a debugger graphical user interface. As per claim 1.

#### **Claim 19**

A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 18 includes the steps of identifying program load and unload events, dynamically updating said program loadmap and displaying said updated loadmap responsive to identified program load and unload events. As per claim 10.

#### **Claim 20**

A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 17 wherein the step of utilizing a custom record display manager for receiving user inputs and implementing a custom record display function includes the steps of identifying a user selected variable and identifying user selected fields of the variable to be displayed, creating a custom record with the user selected fields; and displaying the variable only with the user selected fields. As per claim 6.

#### **Claim 21**

A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 20 includes the steps of identifying a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields. As per claim 7.



***Response to Arguments***

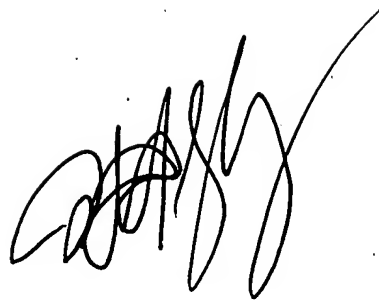
7. Applicant's arguments with respect to claims 1 - 21 have been considered but are moot in view of the Examiner's question of the following. The Examiner is interested in hearing how the added limitations (say Claim 1) are not inherent.

***Correspondence Information***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Todd Ingberg whose telephone number is (571) 272-3723. The examiner can normally be reached on during the work week..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

A handwritten signature in black ink, appearing to read 'Todd Ingberg', with a long, sweeping horizontal stroke extending to the right.

Todd Ingberg  
Primary Examiner  
Art Unit 2193

TI